

Claims

1. A method for processing data packets received at a computing system,
the received data packets being received from a networked transmitting computing
5 entity, the method comprising:

receiving a data packet;

processing lower layer protocol headers of the data packet to expose overlying
headers of the data packet;

10 processing the overlying headers in a shared hardware component capable of
executing header data for a transmission control protocol (TCP) communication and a
storage transport protocol (STP) communication, the header data for the TCP
communication and the STP communication being positioned into standard header
field locations; and

15 determining whether the data packet is from the TCP communication or the
STP communication;

if the data packet is from the TCP communication, completing the
processing of the overlying headers of the data packet separately in TCP
processing;

20 if the data packet is from the STP communication, completing the
processing of the overlying headers of the data packet separately in STP
processing.

2. A method for processing data packets received at a computing system
as recited as recited in claim 1, wherein the header data includes,

checksum data;
data offset data;
compatible flags data;
option types data; and
5 state machine time out data.

3. A method for processing data packets received at a computing system
as recited as recited in claim 2, wherein processing the state machine time out data
includes processing the state machine time out data in an additional shared hardware
10 component used for sending the data packets.

4. A method for processing data packets received at a computing system
as recited as recited in claim 1, further comprising:

transmitting the processed data packet to a buffer of the computing system
15 after completing the processing of the overlying headers of the data packet in the STP
processing or the TCP processing.

5. A method for processing data packets received at a computing system
as recited as recited in claim 1, wherein if the data packet is from the TCP
20 communication, the TCP processing is completed in a TCP hardware unit.

6. A method for processing data packets received at a computing system
as recited as recited in claim 1, wherein if the data packet is from the TCP
communication, the TCP processing is completed by software in a host CPU of the
25 computing system.

7. A method for processing data packets received at a computing system as recited as recited in claim 1, wherein if the data packet is from the STP communication, the STP processing is completed in an STP hardware unit.

5

8. A method for processing data packets received at a computing system, the received data packets being received from a networked transmitting computing entity, the method comprising:

receiving a data packet;

10 processing lower layer protocol headers of the data packet to expose overlying headers of the data packet;

processing the overlying headers in a shared hardware component capable of executing fully compatible header data for a transmission control protocol (TCP) communication and a storage transport protocol (STP) communication, the fully compatible header data for the TCP communication and the STP communication being positioned into standard header field locations;

processing the overlying headers in the shared hardware component capable of executing partially compatible header data for a transmission control protocol (TCP) communication and a storage transport protocol (STP) communication, the partially compatible header data for the TCP communication and the STP communication being positioned into the standard header field locations; and

determining whether the data packet is from the TCP communication or the STP communication;

if the data packet is from the TCP communication, completing the processing of the overlying headers of the data packet separately in TCP processing;

5 if the data packet is from the STP communication, completing the processing of the overlying headers of the data packet separately in STP processing.

9. A method for processing data packets received at a computing system as recited as recited in claim 8, wherein the fully compatible header data includes,

10 checksum data;
data offset data;
compatible flags data;
option types data; and
state machine time out data.

15

10. A method for processing data packets received at a computing system as recited as recited in claim 2, wherein the partially compatible data header data include,

sequence and acknowledgement field data;
20 urgent pointer and urgent flag data;
acknowledgement flag data; and
SRC/DST port and handle data.

12. A method for processing data packets received at a computing system as recited as recited in claim 9, wherein processing the state machine time out data may include processing the state machine time out data in an additional shared hardware component used for sending the data packets.

5

13. A method for processing data packets received at a computing system as recited in claim 8, further comprising:

transmitting the processed data packet to a buffer of the computing system after completing the processing of the overlying headers of the data packet in the STP processing or a TCP processing.

10

14. A method for processing data packets received at a computing system as recited as recited in claim 8, wherein if the data packet is from the TCP communication, the TCP processing is completed in a TCP hardware unit.

15

15. A method for processing data packets received at a computing system as recited as recited in claim 8, wherein if the data packet is from the TCP communication, the TCP processing is completed by software in a host CPU of the computing system.

20

16. A method for processing data packets received at a computing system as recited as recited in claim 8, wherein if the data packet is from the STP communication, the STP processing is completed in an STP hardware unit.

6
17. A method for processing data packets received at a computing system,
the received data packets being received from a networked transmitting computing
entity, the method comprising:

receiving a data packet;

5 processing lower layer protocol headers of the data packet to expose overlying
headers of the data packet;

processing the overlying headers in a shared hardware component capable of
executing header data for a transmission control protocol (TCP) communication and a
storage transport protocol (STP) communication, the header data for the TCP
10 communication and the STP communication being positioned into standard header
field locations; and

determining whether the data packet is from the TCP communication or the
STP communication;

if the data packet is from the TCP communication, completing the
15 processing of the overlying headers of the data packet separately in TCP
processing;

if the data packet is from the STP communication, completing the
processing of the overlying headers of the data packet separately in STP
processing; and

20 transmitting the processed data packet to a buffer of the computing system
after completing the processing of the overlying headers of the data packet in the STP
processing or the TCP processing.

7
18. A method for processing data packets received at a computing system
25 as recited in claim 17, wherein the header data includes,

checksum data;
data offset data;
compatible flags data;
option types data; and
state machine time out data.

19. A method for processing data packets received at a computing system as recited in claim 17, wherein if the data packet is from the TCP communication, the TCP processing is completed in a TCP hardware unit.

20. A method for processing data packets received at a computing system as recited in claim 17, wherein if the data packet is from the TCP communication, the TCP processing is completed by software in a host CPU.

21. A method for processing data packets of multiple formats, comprising processing a first format of data packet header for a first data transfer protocol, the first format having a first plurality of header fields; and

processing a second format of data packet header for a second data transfer protocol, the second format having a second plurality of header fields and being aligned with the first plurality of header fields of the first format of data packet header;

wherein certain ones of the first plurality of header fields and certain ones of the second plurality of header fields are processed with a shared hardware without additional hardware specific for the first data transfer protocol and the second data transfer protocol.

21

22. A method for processing data packets of multiple formats as recited in claim 21, wherein the first data transfer protocol is TCP.

22

5

23. A method for processing data packets of multiple formats as recited in claim 22, wherein the second data transfer protocol is STP.

1066767 "GBF.600F